CG基礎 第六回レポート

201811395 山本雄太

課題(1)について、Bスプライン曲線の実装を行おうとしたが上手くいかなかった。

以下に作成したソースコードと実行結果(失敗例)を示す。

|  |
| --- |
| // 基底関数 N{i,n}(t)の値を計算する  double getBaseN(int i, int n, double t) {  if (n == 0) {  // n が 0 の時だけ t の値に応じて 0 または 1 を返す  if (t >= g\_NotVector[i] && t < g\_NotVector[i + 1]) {  return 1.0;  }  return 0;  }  else {  //☆実装部分☆  double w1 = 0.0, w2 = 0.0;  w1 = ((t - g\_NotVector[i]) / (g\_NotVector[i + n] - g\_NotVector[i])) \* getBaseN(i, n - 1, t);  w2 = ((g\_NotVector[i + n + 1] - t) / (g\_NotVector[i + n + 1] - g\_NotVector[i + 1])) \* getBaseN(i + 1, n - 1, t);  return (w1 + w2);  }  }  // 表示部分をこの関数で記入  void display(void) {  glClearColor(1.0, 1.0, 1.0, 1.0); // 消去色指定  glClear(GL\_COLOR\_BUFFER\_BIT); // 画面消去  // 制御点の描画  glPointSize(5);  glColor3d(0.0, 0.0, 0.0);  glBegin(GL\_POINTS);  for (unsigned int i = 0; i < g\_ControlPoints.size(); i++) {  glVertex2d(g\_ControlPoints[i].x, g\_ControlPoints[i].y);  }  glEnd();  // 制御点を結ぶ線分の描画  glColor3d(1.0, 0.0, 0.0);  glLineWidth(1);  glBegin(GL\_LINE\_STRIP);  for (unsigned int i = 0; i < g\_ControlPoints.size(); i++) {  glVertex2d(g\_ControlPoints[i].x, g\_ControlPoints[i].y);  }  glEnd();  // ★ ここにBスプライン曲線を描画するプログラムコードを入れる  // ヒント1: 3次Bスプラインの場合は制御点を4つ入れるまでは何も描けない  // ヒント2: パラメータtの値の取り得る範囲に注意  glColor3d(0.0, 0.0, 1.0);  glLineWidth(2);  glBegin(GL\_LINE\_STRIP);  double x[100];  double y[100];  for (unsigned int i = 0; i < g\_ControlPoints.size(); i++) {  x[i] = g\_ControlPoints[i].x;  y[i] = g\_ControlPoints[i].y;  }  if (g\_ControlPoints.size() == 3) {  double xp = 0.0, yp = 0.0;  for (int i = 0; i < g\_ControlPoints.size(); i++) {  double r = getBaseN(i, 3, g\_NotVector[3]);  xp = xp + (r \* x[i]);  yp = yp + (r \* y[i]);  }  glVertex2d(xp, yp);  }  else if (g\_ControlPoints.size() > 3) {  double xp = 0.0, yp = 0.0;  for (int t = g\_NotVector[g\_ControlPoints.size()] \* 10; t <= g\_NotVector[g\_ControlPoints.size() + 1]\*10; t++) {  double t1 = (double)t / 10;  for (int i = 0; i < g\_ControlPoints.size(); i++) {  double r = getBaseN(i, 3, t1);  xp = xp + (r \* x[i]);  yp = yp + (r \* y[i]);  glVertex2d(xp, yp);  }  }  }  glEnd();  glutSwapBuffers(); |

実行結果

